

**Amendments to the Specification:**

**Please replace the paragraph beginning at page 1, line 5, with the following rewritten paragraph:**

U.S. Patent Application No. 10/684,272 to Crow et al., filed concurrently herewith, having attorney docket no. 5437-65503, and entitled "CONCEPTUALIZATION OF JOB CANDIDATE INFORMATION" is hereby incorporated by reference herein.

**Please replace the paragraph beginning at page 5, line 30, with the following rewritten paragraph:**

A conceptualizer 142 analyzes ~~structured~~ the structured job candidate data 122 to generate conceptualized job candidate data 152. The conceptualized job candidate data 152 includes one or more concepts extracted (e.g., identified) via analysis of the job candidate data 122. The same concept can be extracted from the job candidate data 122 in a variety of ways. For example, because two candidates may describe the same concept using different language, the same concept may be extracted from two different resumes even though the same language does not appear in the resumes. For example, the concept can be extracted if language somehow denoted as related to a concept is found. For instance, a resume describing a candidate as a "VOIP Engineer" and another resume describing another candidate as a "PBX Engineer" can be represented in software by the same concept.

**Please replace the paragraph at page 5, line 22 with the following rewritten paragraph:**

~~Example 1~~— Exemplary Overview of Exemplary Conceptualization System

**Please replace the paragraph at page 6, line 11 with the following rewritten paragraph:**

~~Example 2~~— Exemplary Overview of Conceptualization Method

**Please replace the paragraph at page 7, line 1 with the following rewritten paragraph:**

~~Example 3~~— Exemplary Overview of Matching System

**Please replace the paragraph at page 7, line 20 with the following rewritten paragraph:**

~~Example 4~~— Exemplary Matching Method

**Please replace the paragraph at page 7, line 28 with the following rewritten paragraph:**

~~Example 5~~— Exemplary Conceptualizer

**Please replace the paragraph at page 8, line 25 with the following rewritten paragraph:**

~~Example 6~~— Exemplary Concepts

**Please replace the paragraph at page 9, line 4 with the following rewritten paragraph:**

~~Example 7~~— Exemplary Ontology

**Please replace the paragraph at page 9, line 27 with the following rewritten paragraph:**

~~Example 8~~— Exemplary Method for Extracting Concepts via an Ontology

**Please replace the paragraph at page 10, line 10 with the following rewritten paragraph:**

~~Example 9~~— Exemplary Ontology Extractor

**Please replace the paragraph at page 10, line 18 with the following rewritten paragraph:**

~~Example 10~~— Exemplary Ontology-Independent Heuristic Extractor and Method

**Please replace the paragraph at page 10, line 29 with the following rewritten paragraph:**

~~Example 11~~— Exemplary Concept Scoring

**Please replace the paragraph at page 11, line 17 with the following rewritten paragraph:**

~~Example 12~~— Exemplary System for Matching via *N*-Dimensional Space

**Please replace the paragraph at page 14, line 8 with the following rewritten paragraph:**

~~Example 13~~— Exemplary Closest Matches

**Please replace the paragraph at page 14, line 27 with the following rewritten paragraph:**

~~Example 14~~—Exemplary Concept Scoring Calculation

**Please replace the paragraph at page 16, line 21 with the following rewritten paragraph:**

~~Example 15~~—Special Organizations

**Please replace the paragraph at page 17, line 8 with the following rewritten paragraph:**

~~Example 16~~—Exemplary Trusted and Speculative Concept Extractors

**Please replace the paragraph at page 17, line 29 with the following rewritten paragraph:**

~~Example 17~~— Exemplary Taxonomy

**Please replace the paragraph at page 18, line 7 with the following rewritten paragraph:**

~~Example 18~~— Exemplary Taxonomy Arrangement: Roles

**Please replace the paragraph at page 18, line 26 with the following rewritten paragraph:**

~~Example 19~~— Exemplary Ontology Entry: “Voice Engineer” Role

**Please replace the paragraph at page 20, line 1 with the following rewritten paragraph:**

~~Example 20~~— Exemplary Extraction Techniques via Ontology

**Please replace the paragraph at page 21, line 4 with the following rewritten paragraph:**

~~Example 21~~— Exemplary Ontology Extractors

**Please replace the paragraph at page 21, line 11 with the following rewritten paragraph:**

~~Example 22~~— Exemplary Parent Ontology Extractor

**Please replace the paragraph at page 22, line 7 with the following rewritten paragraph:**

~~Example 23~~— Exemplary Execution of Parent Ontology Extractor

**Please replace the paragraph at page 20, line 28 with the following rewritten paragraph:**

Second, the ontology allows normalization of the language that job candidates use to express themselves. Whether the candidate’s resume states “Voice Engineer,” “VOIP Engineer,” or “PBX Engineer,” the software can recognize that all ~~there~~ three are alternative ways of expressing the same concepts “Voice Engineer.” By extracting the same concept ‘Role\_Voice Engineer’ regardless of the term used, the system reliably identifies Voice Engineers, even if they do not use the phrase “Voice Engineer” in their resume.

**Please replace the paragraph at page 23, line 8 with the following rewritten paragraph:**

~~Example 24~~— Exemplary Skills Taxonomy

**Please replace the paragraph at page 23, line 16 with the following rewritten paragraph:**

~~Example 25~~— Learning System

**Please replace the paragraph at page 24, line 23 with the following rewritten paragraph:**

~~Example 26~~— Exemplary Execution of Learning System

**Please replace the paragraph at page 26, line 19 with the following rewritten paragraph:**

~~Example 27~~— Exemplary Ontology-independent Heuristic Extractors

**Please replace the paragraph at page 28, line 1 with the following rewritten paragraph:**

~~Example 28~~— Exemplary Ontology-independent Heuristic Extractor: Skills List Extractor

**Please replace the paragraph at page 28, line 15 with the following rewritten paragraph:**

~~Example 29~~— Exemplary Ontology-independent Heuristic Extractor: Skills List Heuristic Extractor Execution

**Please replace the paragraph at page 31, line 11 with the following rewritten paragraph:**

Those sentences declared to be a skills list are then processed to extract skills therefrom. To extract the skills, the following technique can be applied as a particular ~~exemplary implementation~~ exemplary implementation of the method 2100 of FIG. 21:

**Please replace the paragraph at the beginning of page 32 with the following rewritten paragraph:**

~~Example 30~~— Exemplary Ontology-independent Heuristic Extractor: Title Heuristic Extractor

**Please replace the paragraph at page 32, line 26 with the following rewritten paragraph:**

3. Replace common parsing artifacts. For example, “Project & Product Manager” becomes “~~Product~~ Project and Product Manager.”

**Please replace the paragraph at page 33, line 8 with the following rewritten paragraph:**

~~Example 31~~— Exemplary Ontology-independent Heuristic Extractor:  
Exemplary Management Heuristic Extractor

**Please replace the paragraph at page 33, line 24 with the following rewritten paragraph:**

~~Example 32~~— Execution of Exemplary Management Heuristic Extractor

**Please replace the paragraph at page 34, line 25 with the following rewritten paragraph:**

~~Example 33~~— Exemplary Special Purpose Concepts

**Please replace the paragraph at page 35, line 4 with the following rewritten paragraph:**

~~Example 34~~— Exemplary Integrated Assessment Analysis

**Please replace the paragraph at page 35, line 17 with the following rewritten paragraph:**

~~Example 35~~— Candidate Analytics

**Please replace the paragraph at page 35, line 24 with the following rewritten paragraph:**

~~Example 36~~— Exemplary Analytic: Frequent Job Moves

**Please replace the paragraph at page 36, line 4 with the following rewritten paragraph:**

~~Example 37~~— Exemplary Analytic: Career Trajectory Match

**Please replace the paragraph at page 37, line 1 with the following rewritten paragraph:**

~~Example 38~~— Exemplary Matching Functionality

**Please replace the paragraph at page 37, line 13 with the following rewritten paragraph:**

~~Example 39~~— Exemplary System for Generation of Proposed Query Modifications to Control Number of Results Returned by Query

**Please replace the paragraph at page 38, line 4 with the following rewritten paragraph:**

~~Example 40~~— Exemplary Sub-Systems for Generation of Proposed Query Modifications to Control Number of Results Returned by Query

**Please replace the paragraph at page 38, line 6 with the following rewritten paragraph:**

FIG. 25 shows an exemplary system 2500 for proposing query modifications to control the number of results returned by a query. The system can function similarly to the system 2400 of FIG. 24. However, in the example, the forecaster 2532 includes subsystems for proposing dynamic range adjustment 2533, proposing changes to priority 2534, and proposing role-based modifications 2535 to the query 2422. Exemplary implementations of the subsystems are described below.

**Please replace the paragraph at page 38, line 13 with the following rewritten paragraph:**

~~Example 41~~— Exemplary Method for Generation of Proposed Query Modifications to Control Number of Results Returned by Query

**Please replace the paragraph at page 38, line 30 with the following rewritten paragraph:**

At 2720 it is determined whether the number of results (e.g., the number of job candidates returned by the query) is within the desired range. If not, at 2730, it is determined whether the number of results is above the range. If so, at ~~2750~~ 2740, a constraining modification predicted to bring the number of candidates within (or closer to) the range is generated. If not, at ~~2760~~ 2750, a relaxing modification predicted to bring the number of candidates within (or closer to) the range is generated.

**Please replace the paragraph at page 39, line 6 with the following rewritten paragraph:**

~~Example 42~~— Exemplary Implementation of Sub-Systems to Generate Hints

**Please replace the paragraph at page 40, line 1 with the following rewritten paragraph:**

On the other hand, if the generator is generating a relaxing hint, it can identify a component that has the lowest number of candidates associated with it that is currently required and suggest that the identified component be changed to not required.

**Please replace the paragraph at page 40, line 10 with the following rewritten paragraph:**

~~Example 43~~— Exemplary Automated Application of Proposed Query Modifications

**Please replace the paragraph at page 40, line 22 with the following rewritten paragraph:**

~~Example 44~~— Exemplary Cloning



**Please replace the paragraph at page 41, line 1 with the following rewritten paragraph:**

~~Example 45~~— Exemplary Cloning Techniques

**Please replace the paragraph at page 41, line 14 with the following rewritten paragraph:**

Concept selection can be done by a set of five specialized software components (e.g., “cloners” or cloner objects). Each is given the incoming candidate and selects ~~concepts from~~ from concepts to add to the job requisition being constructed. The relative importance of the cloners is configurable. The five cloners can include a role cloner, a skill cloner, a company cloner, an industry cloner, and an education cloner.

**Please replace the paragraph at page 42, line 13 with the following rewritten paragraph:**

~~Example 46~~— Exemplary Architecture for Achieving Matching Functionality

**Please replace the paragraph at page 45, line 9 with the following rewritten paragraph:**

The Freshness field may be set to a special value (e.g., -1) to indicate that candidates with any freshness value can be matched.

**Please replace the paragraph at page 47, line 1 with the following rewritten paragraph:**

~~Example 47~~— Exemplary Data Structures for Achieving Matching Functionality

**Please replace the paragraph at page 49, line 2 with the following rewritten paragraph:**

~~Example 48~~— Exemplary Design for Achieving Matching Functionality via API

**Please replace the paragraph at page 66, line 12 with the following rewritten paragraph:**

~~Example 49~~— Exemplary User Interface Presentation of Match Results

**Please replace the paragraph at page 66, line 25 with the following rewritten paragraph:**

For example, FIG. 31 shows a screenshot of an exemplary graphical user interface 3100 depicting an overview of a candidate (in this case John Smith). In the example, the applicant's name and other information is displayed. In addition, the workstyle match indicator 3140 and thermometer 3145 indicate how well the candidate matches the job workstyle based on a questionnaire (e.g., such as that described in Example 34). Management experience (e.g., the analytic described in Example 31) is also indicated by the indicator 3160. Further, whether the candidate changes jobs frequently (e.g., as described in Example 36) can be indicated by the indicator 3180. Additional, less, or different information can be presented.

**Please replace the paragraph at page 67, line 14 with the following rewritten paragraph:**

~~Example 50~~— Integration into Applicant Tracking Software System

**Please replace the paragraph at page 67, line 9 with the following rewritten paragraph:**

~~Example 51~~— Exemplary Knowledge-Based Human Resources Search

**Please replace the paragraph at page 67, line 16 with the following rewritten paragraph:**

~~Example 52~~— Exemplary Desired Job Candidate Criteria

**Please replace the paragraph at page 67, line 23 with the following rewritten paragraph:**

~~Example 53~~— Exemplary Job Candidates

**Please replace the paragraph at page 68, line 1 with the following rewritten**

**paragraph:**

~~Example 54~~— Exemplary Computer-Readable Media

**Please replace the paragraph at page 68, line 11 with the following rewritten**

**paragraph:**

~~Example 55~~— Exemplary Implementation of Systems